AMENDMENTS TO THE CLAIMS

This listing of claims will replace all prior versions, and listings, of claims in the application.

****Please note the following claims do incorporate the changes made in the Examiner's Amendment, mailed August 16, 2010.****

1. (Previously Presented) A system for parsing an arbitrary input stream, comprising:

a plurality of processor executable parsers operable to parse an input stream, each parser corresponding to a unique set of grammar rules;

a processor executable parser selection agent operable to receive the input stream and select a subset of the plurality of processor executable parsers to parse the input stream, wherein the input stream comprises a plurality of differing input structures and wherein the selected subset of processor executable parsers produce multiple parser outputs; and

a processor executable encoding agent operable to convert the multiple parser outputs to a common grammar, wherein the processor executable parser selection agent and plurality of processor executable parsers are configured in a factory pattern and wherein the input stream comprises a plurality of telecommunication messages from a plurality of telecommunication components, each telecommunication message having a plurality of message headers comprising differing types of information.

2. (Previously Presented) The system of Claim 1, wherein the multiple parser outputs correspond to differing grammars and wherein the multiple parser outputs, as converted into the common grammar, are at least one of further processed and entered into a database for later processing.

3. (Canceled)

4. (Previously Presented) The system of Claim 1, wherein the processor executable parser selection agent is operable to provide to a client, in response to a parse request, at least one of a parser output and an indication when at least some of the input stream is not successfully parsed and wherein the processor executable parser selection agent, prior to

selection of the subset of processor executable parsers, is not informed in advance of the source or input structure associated with the at least some of the input stream.

5. (Canceled)

6. (Previously Presented) A system for parsing an arbitrary input stream, comprising:

a plurality of processor executable parsers operable to parse an input stream, each processor executable parser corresponding to a unique input structure;

a processor executable parser selection agent operable to receive the input stream and select a subset of the plurality of processor executable parsers to parse the input stream, wherein the input stream comprises a plurality of differing input structures and wherein the selected subset of processor executable parsers produce multiple parser outputs corresponding to the plurality of differing input structures and differing grammars; and

a processor executable encoding agent operable to convert the multiple parser outputs to a common grammar, wherein the input stream comprises fault information, the fault information being related to at least one of an alarm and an error, the fault information comprising first fault information related to a first event and in a first format and second fault information related to a second event discrete from the first event and in a second format different from the first format and wherein the processor executable encoding agent is operable to convert the first and second formats to a common format, wherein the first and second fault information uses different characters to refer to a same type of event and the encoding agent is further operable to convert the different characters to a common set of characters to refer to the event.

7. (Canceled)

8. (Previously Presented) A method for parsing an arbitrary input stream, comprising:

(a) receiving, from a telecommunication device, an input stream, the input stream comprising information defined by at least first and second input structures;

(b) providing, by a processor executable parser selection agent and substantially simultaneously, a common portion of the input stream to each of a plurality of processor executable parsers, the plurality of processor executable parsers corresponding to differing sets of grammars;

(c) receiving, by the processor executable parser selection agent, output from each of the plurality of processor executable parsers;

(d) based on the plurality of outputs of the plurality of processor executable parsers:

(i) in a first mode selecting, by the processor executable parser selection agent, a first output from a first processor executable parser that corresponds to the first input structure and a second output from a second parser that corresponds to the second input structure; and

(ii) in a second mode, selecting, by the processor executable parser selection agent, a first processor executable parser corresponding to the first input structure to parse one or more first segments of the input stream and a second processor executable parser corresponding to the second input structure to parse one or more second segments of the input stream; and

(e) converting a first parser output from the plurality of processor executable parsers into a second parser output corresponding to a common grammar.

9. (Previously Presented) The method of Claim 8, wherein the first mode (d)(i) is performed.

10. (Previously Presented) The method of Claim 8, wherein the second mode (d)(ii) is performed.

11. (Previously Presented) The method of Claim 8, wherein the input stream comprises a plurality of nonstandardized headers.

12. (Previously Presented) The method of Claim 8, wherein the input stream is free of an embedded tag indicating a source and/or input structure associated with the input stream and wherein step (b) comprises:

identifying one or more tokens in the input stream; and

based on the identified one or more tokens, selecting the at least one of a plurality of processor executable parsers.

13. (Currently Amended) A method for parsing an arbitrary input stream, comprising:

(a) receiving an input stream, the input stream comprising information defined by at least first and second input structures;

(b) providing, substantially simultaneously, a common portion of the input stream to each of a plurality of processor executable parsers, the plurality of processor executable parsers corresponding to differing sets of grammars, wherein step (b) comprises:

determining, by a processor executable parser selection agent, for each of the at least one of a plurality of processor executable parsers whether a match or a ~~not match~~no match condition exists, a match condition indicating that a selected processor executable parser has successfully parsed a selected segment of the input stream and a no match condition indicating that the selected processor executable parser has not successfully parsed the selected segment of the input stream; and

applying, by the processor executable parser selection agent, the following rules:

when, for a parsed segment, only one match condition is found to exist, not generating an error message;

when, for a parsed segment, a match condition is not found to exist, generating an error message; and

when, for a parsed segment, multiple match conditions are found to exist, generating an error message;

(c) receiving output from each of the plurality of processor executable parsers; and

(d) based on the outputs of the plurality of processor executable parsers, performing, by the processor executable parser selection agent, at least one of:

(i) selecting a first output from a first processor executable parser that corresponds to the first input structure and a second output from a second parser that corresponds to the second input structure; and

(ii) selecting a first processor executable parser corresponding to the first input structure to parse one or more first segments of the input stream and a second processor executable parser corresponding to the second input structure to parse one or more second segments of the input stream.

14. (Previously Presented) The method of Claim 9, wherein a third processor executable parser successfully parses a first portion of the input stream to form a third output and the first processor executable parser successfully parses the first portion of the input stream to form a first output and further comprising:

determining, by the processor executable parser selection agent, which of the first and third outputs most likely corresponds to the first portion.

15. (Previously Presented) The method of Claim 14, wherein the determining step is performed using a least squares fit analysis and wherein step (d) is performed using a declarative programming rather than procedural programming approach.

16. (Previously Presented) The method of Claim 8, wherein the first processor executable parser produces a first output and the first output is a parse tree and further comprising:

recursively evaluating at least some of the nodes in the parse tree to identify nodes requiring additional parsing.

17. (Previously Presented) The method of Claim 8, wherein the first processor executable parser produces a first output and the first output is a parse tree and further comprising:

recursively examining at least some of the nodes in the parse tree to identify nodes of interest to a client.

18. (Previously Presented) The method of Claim 8, wherein the first processor executable parser produces a first output and the first output is a parse tree and wherein at least first and second nodes of the parse tree have differing formats and further comprising:

iteratively traversing a plurality of the nodes of the parse tree to locate nodes of interest, the nodes of interest comprising the first and second nodes; and

converting each of the located nodes of interest to a standard format.

19. (Original) The method of Claim 18, wherein each of the first and second nodes use different characters to refer to a same type of event and further comprising:

converting the characters in the first and second nodes to a common set of characters to refer to the type of event.

20. (Previously Presented) The method of Claim 8, wherein each of the plurality of processor executable parsers corresponds to a unique set of tokens and grammar rules.

21. (Previously Presented) The method of Claim 8, wherein each of the plurality of processor executable parsers corresponds to a unique set of attribute grammars.

22. (Previously Presented) A non-transient computer readable medium containing processor executable instructions, wherein a processor executing the instructions performs the steps of Claim 8.

23. (Previously Presented) A method for parsing computer generated information, comprising:

receiving a stream of information, the stream being generated by one of a plurality of possible different telecommunication components, wherein each telecommunication component generates a stream corresponding to a unique input structure and wherein each of a plurality of differently structured segments of the stream is free of an embedded tag indicating a corresponding telecommunication component and/or input structure for the respective segment;

comparing, by a processor executable heuristic parser, at least a portion of the stream with multiple different sets of tokens to provide a subset of tokens identified in the at least a portion of the stream, each set of tokens corresponding to a unique input structure;

based on the subset of tokens, heuristically identifying, by the processor executable heuristic parser and from among at least one of a plurality of possible input structures and a plurality of possible telecommunication components, at least one of an input structure corresponding to the at least a portion of the stream and a telecommunication component for the at least a portion of the stream; and

thereafter parsing the stream based on the identified at least one of an input structure and telecommunication component.

24. (Previously Presented) The method of Claim 23, wherein the input stream comprises a plurality of headers, wherein the headers comprise differing types of information, wherein each of the tokens has a corresponding method expressing a set of syntactical and/or semantical relationships relating to the respective token and wherein the heuristically identifying step comprises:

for each token in the subset of tokens, invoking a corresponding method.

25. (Previously Presented) The method of Claim 24, wherein the comparing and heuristically identifying steps are performed using a declarative programming approach rather than a procedural programming approach, wherein the headers are nonstandardized, and wherein the invoking step comprises

setting, by an invoked method, a set of flags depending on the presence or absence of a syntactical and/or semantical relationship; and wherein the values of the flags are used to heuristically identify the at least one of an input structure corresponding to the at least a portion of the stream and a telecommunication component for the at least a portion of the stream.

26. (Previously Presented) The method of Claim 23, wherein the stream of information comprises graphical information, wherein the comparing, heuristically identifying, and parsing steps are performed by a processor executable parser, wherein the processor executable parser is not provided with a flag external to the input stream to identify or assist in the identification of the at least one of an input structure corresponding to the at least a portion of the stream and a telecommunication component for the at least a portion of the stream.

27. (Previously Presented) A non-transient computer readable medium containing processor executable instructions, wherein, when the instructions are executed by a processor, the processor performs the steps of Claim 23.

28. (Previously Presented) A system, comprising:

an input operable to receive a stream of information, the stream being generated by one of a plurality of possible different telecommunication components, wherein each telecommunication component generates a stream corresponding to a unique input structure; and

a processor executable parser operable to:

(a) compare at least a portion of the stream with multiple different tokens to provide a subset of tokens identified in the at least a portion of the stream, each token corresponding to a unique input structure;

(b) based on the subset of tokens, identify, from among at least one of a plurality of possible input structures and a plurality of possible telecommunication components, at least one of an input structure corresponding to the at least a portion of the stream and a telecommunication component for the at least a portion of the stream; and

(c) parse the stream based on the identified at least one of an input structure and telecommunication component, wherein the processor executable parser is not provided with an input structure identifier, other than the corresponding input structure itself, either in or external to the at least a portion of the input stream to identify or assist in the identification of the at least one of the respective input structure corresponding to the at least a portion of the stream and a telecommunication component for the at least a portion of the stream.

29. (Previously Presented) The system of Claim 28, wherein each of the tokens has a corresponding parser expressing a set of syntactical and/or semantical relationships relating to the respective token and wherein the processor executable parser is further operable, for each token in the subset of tokens, to (d) to invoke a corresponding method.

30. (Previously Presented) The system of Claim 29, wherein the processor executable parser is further operable to (e) assign, by an invoked method, a set of flags with a corresponding set of values depending on the presence or absence of a syntactical and/or semantical relationship, wherein the values of the flags are used to heuristically identify the at least one of an input structure corresponding to the at least a portion of the stream and a telecommunication component for the at least a portion of the stream.

31. (Previously Presented) The system of Claim 28, wherein the stream of information comprises graphical information, wherein the processor executable parser is not provided, by another telecommunication component, with a flag external to the input stream to identify or assist in the identification of the at least one of an input structure corresponding to the at least a portion of the stream and a telecommunication component for the at least a portion of the stream.